

L Number	Hits	Search Text	DB	Time stamp
-	2	("20020029231").PN.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 19:33
-	2260	documentation same (source adj code)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 11:49
-	12	(documentation same (source adj code)) same textual	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 08:58
-	45	documentation same (source adj code) same debugger	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 11:51
-	0	documentation same (source adj code) same (diagram or graph) same debugger	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 11:51
-	70	documentation same (source adj code) same (diagram or graph)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 11:52
-	2	("5530942").PN.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 14:45
-	4245	version near4 control	USPAT	2004/04/28 14:45
-	3220	(version near4 control) and diagram	USPAT	2004/04/28 14:45
-	280	(version near4 control) same diagram	USPAT	2004/04/28 14:45
-	1	((version near4 control) same diagram) same (source adj code)	USPAT	2004/04/28 14:55
-	9	dreamweaver	USPAT	2004/04/28 15:03
-	2	cvs same diagram same (source adj code)	USPAT	2004/04/28 15:06
-	42	cvs same diagram	USPAT	2004/04/28 15:06
-	71	language-neutral	USPAT	2004/04/28 19:33
-	71	language-neutral	USPAT	2004/04/28 20:06
-	0	language-neutral same (source adj code) same (diagram or graph or image)	USPAT	2004/04/29 10:10
-	4	language-neutral same (source adj code) same (diagram or graph or image)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/28 20:07
-	9	((("5940615") or ("4488256") or ("5818711") or ("5530942") or ("5905508") or ("5592600") or ("5813019") or ("5841959") or ("5526520")).PN.	USPAT	2004/04/29 10:07
-	0	language-neutral same (transient adj meta adj model)	USPAT	2004/04/29 10:10
-	14	language-neutral same (transient adj meta adj model)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 10:10
-	10	(US-5530942-\$ or US-5940615-\$ or US-5905508-\$ or US-5841959-\$ or US-5818711-\$ or US-5813019-\$ or US-5592600-\$ or US-5526520-\$ or US-4488256-\$).did. or (US-20020029231-\$).did.	USPAT; US-PGPUB	2004/04/29 14:05

-	0	((US-5530942-\$ or US-5940615-\$ or US-5905508-\$ or US-5841959-\$ or US-5818711-\$ or US-5813019-\$ or US-5592600-\$ or US-5526520-\$ or US-4488256-\$).did. or (US-20020029231-\$).did.) and ((image adj map) same diagram)	USPAT	2004/04/29 14:07
-	2	((US-5530942-\$ or US-5940615-\$ or US-5905508-\$ or US-5841959-\$ or US-5818711-\$ or US-5813019-\$ or US-5592600-\$ or US-5526520-\$ or US-4488256-\$).did. or (US-20020029231-\$).did.) and ((image adj map))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:09
-	210	((image adj map) same HTML)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:09
-	5	((image adj map) same HTML same diagram)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:47
-	210	((image adj map) same HTML same image)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:53
-	10	((image adj map) same HTML same image same gif)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:50
-	10	((image adj map) same HTML same gif)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:55
-	61	((image adj map) same HTML same (region or area))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 14:55



US Patent & Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

rational rose



THE ACM DIGITAL LIBRARY



[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used **rational rose**

Found **5,804** of **132,857**

Sort results by



[Save results to a Binder](#)

Try an [Advanced Search](#)

Display results



[Search Tips](#)

Try this search in [The ACM Guide](#)

☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Automatically connecting documentation to code with rose](#)

Robert Pierce, Scott Tilley

October 2002 **Proceedings of the 20th annual international conference on Computer documentation**

Full text available: pdf(256.32 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

One of the most common problems with program documentation is keeping it synchronized with the source code it purports to explain. One solution to this problem is to automate the documentation process using reverse engineering technology. Reverse engineering is an emerging branch of software engineering that focuses on recreating high-level information (such as program documentation) from low-level artifacts (such as source code). This paper describes an automated approach to maintaining the con ...

Keywords: application programming interface (API), automation, documentation, single sourcing, software engineering

2 [Developing a user information architecture for Rational's ClearCase product family documentation set](#)

Mary Hunter Utt, Robert Mathews

October 1999 **Proceedings of the 17th annual international conference on Computer documentation**

Full text available: pdf(822.09 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Information architecture, like information development and delivery, has much in common with its software counterpart. This paper describes how the Rational ClearCase® documentation group developed an information architecture to meet changing industry, corporate, and product requirements. During this work, it became clear that our architecture development process mapped closely to the Rational Unified Process, an iterative and incremental approach to software architecture and developmen ...

Keywords: ClearCase documentation, RUP, Rational Unified Process, information architecture

3 [Software process improvement by object technology \(ESSI PIE 27785 — SPOT\)](#)

Antonio Cagliò, Massimo Autiero, Giuseppe Bux

Full text available:  [pdf\(228.50 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper describes the on going experience of Calio Informatica Srl in a project of Process Improvement Experiment (PIE) sponsored by the Community's ESSI (European Systems and Software Initiative) program. The experiment concerns the improvement of two primary Software Life Cycle (SLC) processes, namely Analysis and Design, by adopting Object Oriented technology, and in particular the UML method. Rational Rose is the technology, which is supporting the improvement. The PIE is bein ...

Keywords: Rational Rose, object oriented technology, software components, software reuse, three-tier reference model, unified modelling language UML

4 Modelling: Reveal: a tool to reverse engineer class diagrams

Sarah Matzko, Peter J. Clarke, Tanton H. Gibbs, Brian A. Malloy, James F. Power, Rosemary Monahan

February 2002 **Proceedings of the Fortieth International Confernece on Tools Pacific: Objects for internet, mobile and embedded applications - Volume 10**

Full text available:  [pdf\(1.00 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


Many systems are constructed without the use of modeling and visualization artifacts, due to constraints imposed by deadlines or a shortage of manpower. Nevertheless, such systems might profit from the visualization provided by diagrams to facilitate maintenance of the constructed system. In this paper, we present a tool, Reveal, to reverse engineer a class diagram from the C + + source code representation of the software. In Reveal, we remain faithful to the UML standard definition of a ...

Keywords: UML, automated construction, class diagram, object-oriented programming, reverse engineering, unified modeling language

5 A simple method for drawing a rational curve as two Bézier segments

Jean Gallier

October 1999 **ACM Transactions on Graphics (TOG)**, Volume 18 Issue 4

Full text available:  [pdf\(138.32 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)


In this paper we give a simple method for drawing a closed rational curve specified in terms of control points as two Bézier segments. The main result is the following: For every affine frame (r,s) (where $r < s$), for every rational curve $F(t)$ specified over $[r,s]$ by some control polygon $(\&bgr;0, \dots, \&bgr;m)$ (where the $\&bgr$...

Keywords: Bézier curves, control points, de Casteljau algorithm, rational curves, subdivision, weights

6 Preliminary defect data from the iterative development of a large C++ program (experience report)

James F. Walsh

October 1992 **ACM SIGPLAN Notices , conference proceedings on Object-oriented programming systems, languages, and applications**, Volume 27 Issue 10

Full text available:  [pdf\(644.66 KB\)](#) Additional Information: [full citation](#), [references](#), [citing](#), [index terms](#)

In support of user interface design in the rational unified process

Chris Phillips, Elizabeth Kemp

January 2002 **Australian Computer Science Communications , Third Australasian conference on User interfaces - Volume 7**, Volume 24 Issue 4

Full text available:  [pdf\(509.24 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The Rational Unified Process (RUP) is a use case driven iterative software engineering process User Interface design within the RUP involves user interface modelling and user interface prototyping. This paper describes two support artefacts - extended tabular use cases and UI element clusters - which provide a bridge between these two activities. They provide support for 'flow of events' storyboarding, the clustering of user interface elements and identification of UML boundary classes, and the ...

Keywords: UML, rational unified process, use case modelling, user interface design

8 Design patterns and real-time object-oriented modeling (poster session)

Ross McKegney, Terry Shepard

January 2000 **Addendum to the 2000 proceedings of the conference on Object-oriented programming, systems, languages, and applications (Addendum)**

Full text available:  [pdf\(22.58 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Real-time object-oriented modeling tools (e.g. Rational Rose-RT, i-Logix Rhapsody) allow developers to focus on software architecture by abstracting away low-level implementation details. We believe that design patterns can be very beneficial in this context, and present the rationale and concepts behind a proposal for the extension of such a toolset to support them explicitly.

9 Reuse in a simulation environment for flexible manufacturing systems

Sándor Kopácsi, Daniela Gavalocová, George L. Kovács, Ildikó Kmecs

September 1996 **ACM SIGAPP Applied Computing Review**, Volume 4 Issue 2

Full text available:  [pdf\(529.81 KB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

Software developers often meet the problem of creating new components of an application that someone probably has already produced. Without having effective reuse tools usually it is more natural to create new components from scratch than to seek for useful elements in other programmes and/or systems. In the field of simulation of Flexible Manufacturing Systems (FMS) this issue often occurs, when different systems with some similar features have to be managed. FMSs are highly automated production ...

Keywords: G2, SIMAN, flexible, manufacturing systems, modelling, simulation, software reuse

10 Software architecture—a rational metamodel

Philippe Kruchten

October 1996 **Joint proceedings of the second international software architecture workshop (ISAW-2) and international workshop on multiple perspectives in software development (Viewpoints '96) on SIGSOFT '96 workshops**

Full text available:  [pdf\(315.04 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

11

Increasing the flexibility of modelling tools via constraint-based specification

Philip Gray, Ray Welland

November 1999 **Proceedings of the 1999 conference of the Centre for Advanced Studies on Collaborative research**


Full text available:  pdf(87.96 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Most commercial modelling tools provide support for customising surface features (i.e., visual and, to some extent, interactive behaviour) of a model. Although useful and simple to use, such customisation is typically very limited; for example, one cannot change the basic representation of model components. Meta-CASE tools offer the potential for much greater customisation, but at a high cost, viz., the tool must be respecified and rebuilt. We propose an approach, constraint-based specification, ...

12 Unifying documentation teams

Karl A. Hakkarainen

October 1999 **Proceedings of the 17th annual international conference on Computer documentation**

Full text available:  pdf(1.11 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper describes how a set of geographically and organizationally distributed documentation teams created the Rational Suite 1.0 documentation set. The paper covers the business operations of Rational Software, details the documentation tools and technologies used in the project and describes the evolution of the larger team as it learned how to work with a new software development methodology. The paper concludes with a summary of lessons learned and next steps.

Keywords: distributed teams, documentation standards, organizational development

13 Design: Designing UML diagrams for technical documentation

Neil MacKinnon, Steve Murphy

October 2003 **Proceedings of the 21st annual international conference on Documentation**

Full text available:  pdf(276.00 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


This paper presents a framework for improving the presentation of Unified Modeling Language (UML) diagrams, as applied in technical documentation produced at the IBM Toronto Software Laboratory. UML diagrams are a key part of program design. They can enhance understanding of complex programming concepts, and assist in problem analysis and solution design. In turn, UML diagrams can add significant value to documentation, helping the user to understand not only the solution but also the reasons for ...

Keywords: UML diagrams, documentation, graphic design, guidelines, human factors, unified modeling language, visualization

14 Mapping UML designs to Java

William Harrison, Charles Barton, Mukund Raghavachari

October 2000 **ACM SIGPLAN Notices , Proceedings of the 15th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 35 Issue 10

Full text available:  pdf(133.74 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Tools for the generation of code from model descriptions are valuable in helping developers maintain consistency between a model and its implementation. In this paper, we describe a new method for generating Java implementation code from UML diagrams. Our method accepts UML diagrams specified at a higher-level than current tools, and places fewer

constraints on the supported UML constructs, like multiple generalizations and association classes. Unlike current tools, it generates implementation c ...

Keywords: Java, UML, code generation, design, separation of concerns

15 Automated abstraction of class diagrams

Alexander Egyed

October 2002 **ACM Transactions on Software Engineering and Methodology (TOSEM)**,

Volume 11 Issue 4

Full text available:  pdf(1.76 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Designers can easily become overwhelmed with details when dealing with large class diagrams. This article presents an approach for automated abstraction that allows designers to "zoom out" on class diagrams to investigate and reason about their bigger picture. The approach is based on a large number of abstraction rules that individually are not very powerful, but when used together, can abstract complex class structures quickly. This article presents those abstraction rules and an algorithm for ...



Keywords: Class abstraction, class diagrams, class patterns, reverse engineering, transformation, unified modeling language

16 From use cases to code - rigorous software development with UML

Albert Zündorf

July 2001 **Proceedings of the 23rd international conference on Software engineering**

Full text available:

 pdf(49.03 KB) 

[Publisher Site](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The Rational Unified Process lacks technical guidance for the development of object oriented applications. This tutorial fills this gap. We first use UML scenario diagrams to analyze use-cases. Next, we show a method to analyze scenarios and to derive UML class diagrams and UML behavior modeling for active classes and methods. We show how to choose and embed design patterns in a design and how to employ different architectural styles. From such a precise design, smart CASE tools generate full ...

Keywords: CASE tools, UML, development processes, objectoriented applications, software development methods

17 Case tools in object-oriented analysis and design

Hubert A. Johnson, Laura Wilkinson

December 2003 **The Journal of Computing in Small Colleges**, Volume 19 Issue 2

Full text available:  pdf(354.57 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper which falls in the practice domain describes the utilization of CASE tools in enhancing students understanding of the design process in a software engineering course. This newly designed course is an outgrowth of a previous software engineering course in which all phases of software development were considered. In the previous course students often found it overwhelming in dealing with the software engineering concepts and principles and developing the system at the same time. This ne ...

18 Object-oriented technology: A database representation that improves automation and maintains consistency in a multiple view environment

Fawzi Albalooshi

July 2002 **Proceedings of the 14th international conference on Software engineering**

and knowledge engineering

Full text available:  [pdf\(147.79 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


The paper presents an approach to information representation that takes advantage of the incremental nature of software development to facilitate automation. Software is developed in stages where input to a stage is highly dependent on the stage that precedes it. A canonical database representation that is shared by the development tools is presented. It aims at the provision of maximum automated services in the development process. At the same time ensuring the consistency of design information ...

Keywords: automation, information representation, multiple views, object-orientation, software development

19 UML-Based integration testing

Jean Hartmann, Claudio Imoberdorf, Michael Meisinger

August 2000 **ACM SIGSOFT Software Engineering Notes , Proceedings of the International Symposium on Software Testing and Analysis**, Volume 25 Issue 5

Full text available:  [pdf\(761.34 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Increasing numbers of software developers are using the Unified Modeling Language (UML) and associated visual modeling tools as a basis for the design and implementation of their distributed, component-based applications. At the same time, it is necessary to test these components, especially during unit and integration testing. At Siemens Corporate Research, we have addressed the issue of testing components by integrating test generation and test execution technology with commerci ...

Keywords: COM/DCOM, CORBA, UML statecharts, distributed components, functional testing, test execution, test generation

20 Managing the software design documents with XML

Junichi Suzuki, Yoshikazu Yamamoto

September 1998 **Proceedings of the 16th annual international conference on Computer documentation**

Full text available:  [pdf\(1.09 MB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

Keywords: CASE data interchange, UML, XML, software model interchange

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright ?2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)